

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for creating software, comprising:
providing a plurality of nodes and a directory of applications, each of an application being created by use of at least a portion of the plurality of selecting at least a portion of the plurality of nodes;
selecting at least a portion of plurality of nodes to create a selected node layout that represent a plurality of application logics; ~~and~~
executing the selected node layout by a server program
visually showing the selected node layout as a visual node layout;
monitoring a flow of control through each node in the node layout during execution by
showing individual node execution measurement; and
wherein the individual node execution measurements include usage totals, whole
execution time and average execution time.
2. (Original) The method of claim 1, further comprising:
visually displaying the selected node layout as a visual node layout.
3. (Original) The method of claim 1, wherein at least a portion of the plurality of application logics includes a user interaction.
4. (Original) The method of claim 3, wherein the user interaction permits a user to interact with the server program.
5. (Original) The method of claim 3, wherein the user interaction is executable on multiple channels.

6. (Original) The method of claim 3, wherein the user interaction is executable by at least one of web, voice, e-mail and wireless channels
7. (Original) The method of claim 1, wherein the plurality of nodes includes a user interface node.
8. (Original) The method of claim 7, wherein the user interface node includes GUI components and a template for the physical layout of static and dynamic portions of a user display specific GUI components.
9. (Currently Amended) The method of claim 8, wherein dynamic portions of the user display are used by the server program at runtime to layout application.
10. (Original) The method of claim 3, wherein the user interaction includes a user interface node, a user interface block node and an interaction node.
11. (Original) The method of claim 10, wherein the user interface node and user interface block node create a user interaction based on business rules.
12. (Original) The method of claim 11, wherein the interaction node executes the user interaction.
13. (Original) The method of claim 1, wherein each node is a visual representation of a software function.
14. (Original) The interface of claim 10, wherein each node includes inputs to a software function.
15. (Original) The interface of claim 1, wherein the plurality of nodes includes task node interfaces with external components to exchange data information.

16. (Currently Amended) The method of claim 1, wherein the selected node layout can be debugged visually.
17. [[.]] (Currently Amended) The method of claim 1, wherein the parameter and properties values of the nodes can be changed dynamically based on business rules.
18. (Currently Amended) The method claim 1, wherein the parameter and properties values can be linked to variables.
19. (Original) The method of claim 1, wherein the application logic is directly executed without compilation of application logic.
20. (Currently Amended) The method of claim 1, wherein the application logic can be paused and saved during execution~~exeuotion~~.
21. (Original) The method of claim 21, wherein the saved application logic can be restored and resumed.
22. (Currently Amended) The method of claim ~~2022~~, wherein the saved application logic can be restored and execution resumed on a copy of the server program on a computer other than where it was initially started.
23. (Currently Amended) A method for creating software, comprising:
providing a plurality of nodes and a directory of applications, each of an application being created by use of at least a portion of the plurality of the nodes;
selecting at least a portion of the plurality of nodes to create a selected node layout that represent a plurality of application logics;
defining the application logic by selecting at least one of GUI parameters and options in each selected node; [[.]]
executing the selected node layout by a server program;

visually displaying the selected node layout as a visual node layout;
monitoring a flow of control through each node in the node layout during execution by
displaying individual node execution measurement; and
wherein the individual node execution measurements include usage counts, total
execution time and average execution time.

24. (Canceled)

25. (Canceled)

26. (Canceled)

27. (Original) The method of claim 23, further comprising: providing documentation of a functional use of a node.

28. (Original) The method of claim 23, further comprising: providing a graphic description of a plurality of nodes that represent a full application logic.

29. (Original) The method of claim 23, further comprising:
creating a history of different versions of the application logic.

30. (Original) The method of claim 23, further comprising:
creating access control of the application logic.

31. (Original) The method of claim 30, wherein the access control provides single access of the application logic for purposes of modification and multiple access of the application logic for purposes of viewing.

32. (Original) The method of claim 23, further comprising:
automatically validating the application logic against errors.

33. (Original) The method of claim 23, further comprising:
aggregation at least a portion of the plurality of nodes to create an aggregated node.
34. (Original) The method of claim 33, wherein the aggregated node is an application logic.
35. (Original) The method of claim 34, wherein the aggregated node can be used different application logics.
36. (Currently Amended) A method for creating software, comprising:
providing a plurality of nodes and a directory of applications, each of an application being created by use of at least a portion of the plurality of the nodes;
selecting at least a portion of the plurality of nodes to create a selected node layout that represent a plurality of application logics;
defining external application interfaces; and
executing the selected node layout by a server program.
illustrating the selected node layout as a visual node layout;
observing a flow of control through each node in the node layout during execution by
illustrating individual node execution measurement; and
wherein the individual node execution measurements include usage tallies, entire execution time and mean execution time.
37. (Original) The method of claim 36, further comprising:
establishing conditions for execution of the selected node layout.
38. (Original) The method of claim 37, wherein the conditions for the execution include time based events.
39. (Original) The method of claim 37, wherein the conditions for the execution include programmatic events.

40. (Original) The method of claim 39, wherein selected programmatic events create a trigger for the execution of the selected node layout.